Covid-19 Cases Animated Bar Chart Race

Our task is to create an animated bar chart race(bcr) for the number of countrywise covid-19 cases between the time period of Feb 2020 to April 2021.

Unlike other tutorials that allow you to use a pre loaded bcr dataset, we will process and clean our own dataset. Modify it so that it can be used to create Bar Chart Race

About the problem

Our Problem statement would be Covid-19 cases records around the world.

“Hope is being able to see that there is light despite all of the darkness.” – Desmond Tutu

About Dataset

This data was scraped from woldometers.info on 2021-04-24 by Joseph Assaker.

218 countries are represented in this data.

All of countries have records dating from 2020-2-15 until 2021-04-24 (435 days per country). That's with the exception of China, which has records dating from 2020-1-22 until 2021-04-24 (459 days per country).

Acknowledgements for Dataset

All the data present in this dataset is scraped from worldometers.info.

Load Libraries

In [1]:

```
import pandas as pd
import os
```

Load Dataset

In [2]:

```
df = pd.read_csv("worldometer_coronavirus_daily_data.csv")
```

Processing the Dataset : Let's get to Know the data

```
In [3]:  df.head()
```

Out[3]:

	date	country	cumulative_total_cases	daily_new_cases	active_cases	cumulative_total_deaths	daily_new_deaths
0	2020-2-15	Afghanistan	0.0	NaN	0.0	0.0	NaN
1	2020-2-16	Afghanistan	0.0	NaN	0.0	0.0	NaN
2	2020-2-17	Afghanistan	0.0	NaN	0.0	0.0	NaN
3	2020-2-18	Afghanistan	0.0	NaN	0.0	0.0	NaN
4	2020-2-19	Afghanistan	0.0	NaN	0.0	0.0	NaN

```
In [4]:  df.shape
```

```
Out[4]: (95289, 7)
```

```
In [5]:  df.tail()
```

Out[5]:

	date	country	cumulative_total_cases	daily_new_cases	active_cases	cumulative_total_deaths	daily_new_deaths
95284	2021-4-20	Zimbabwe	37875.0	16.0	1263.0	1554.0	1.0
95285	2021-4-21	Zimbabwe	37980.0	105.0	1360.0	1555.0	1.0
95286	2021-4-22	Zimbabwe	38018.0	38.0	1390.0	1555.0	0.0
95287	2021-4-23	Zimbabwe	38045.0	27.0	1395.0	1556.0	1.0
95288	2021-4-24	Zimbabwe	38064.0	19.0	1407.0	1556.0	0.0

As there are many countries data , we need to Select some Particular Countries Data which we want to analyse

```
In [6]:  df.loc[df["country"] == "Zimbabwe"].shape
```

```
Out[6]: (435, 7)
```

```
In [7]:  df.loc[df["country"] == "India"].shape
```

Out[7]:

```
(435, 7)
```

In [8]:

```
df.loc[df["country"] == "China"].shape
```

Out[8]:

```
(459, 7)
```

Thus, we have values for around 450 days for each country.

In [9]:

```
df.isnull().sum()
```

Out[9]:

```
date                          0
country                       0
cumulative_total_cases        0
daily_new_cases            6469
active_cases                  0
cumulative_total_deaths    6090
daily_new_deaths          19190
dtype: int64
```

Selecting countries for Bar Plot

Picking up the cumulative_total_cases column as series and group them with countries name. I will pick up 8 countries , most populas and our neighbours for evaluation

In [10]:

```
russia = df.loc[df["country"] == "Russia"]["cumulative_total_cases"].reset_index(drop =True)

uk = df.loc[df["country"] == "UK"]["cumulative_total_cases"].reset_index(drop =True)

pakistan = df.loc[df["country"] == "Pakistan"]["cumulative_total_cases"].reset_index(drop =True)

india = df.loc[df["country"] == "India"]["cumulative_total_cases"].reset_index(drop =True)

china = df.loc[df["country"] == "China"]["cumulative_total_cases"].reset_index(drop =True)

bangladesh = df.loc[df["country"] == "Bangladesh"]["cumulative_total_cases"].reset_index(drop =True)

brazil = df.loc[df["country"] == "Brazil"]["cumulative_total_cases"].reset_index(drop =True)

usa = df.loc[df["country"] == "USA"]["cumulative_total_cases"].reset_index(drop =True)
```

In [11]:

```
usa.tail()
```

Out[11]:
```
430    32536920.0
431    32602224.0
432    32669279.0
433    32736373.0
434    32789653.0
Name: cumulative_total_cases, dtype: float64
```

We have data for only 435 rows for all the countries. Hence lets take data for 435 rows for china too.

Processing Data For China

In [12]:
```
CHINA=[]
for i in range(0,435):
    CHINA.append(china[i])
```

Converting to series

In [13]:
```
china = pd.Series(CHINA)
```

In [14]:
```
china.shape
```

Out[14]:
```
(435,)
```

Great! Now the length of all our columns are in sink ! i.e 435 days data.

We also need to pick up the column of date. To retrieve only first 459 values , as the dates are repeatitive with countries , the max days data we have is for 435 days for all the countries.

Processing Date Column:

In [15]:
```
date=[]
for i in range(0,435):
    date.append(df.date[i])
```

Converting list to series:

In [16]:
```
DATE = pd.Series(date)
```

In [17]:
```
DATE
```

```
0      2020-2-15
```

Out[17]:

```
1      2020-2-16
2      2020-2-17
3      2020-2-18
4      2020-2-19
          ...
430    2021-4-20
431    2021-4-21
432    2021-4-22
433    2021-4-23
434    2021-4-24
Length: 435, dtype: object
```

In [18]:

```
india.index
```

Out[18]:

```
RangeIndex(start=0, stop=435, step=1)
```

In [19]:

```
uk.isnull().sum()
```

Out[19]:

```
0
```

Concatenating series to create a new database

As we now have different series let's give name to these series that would later be converted to Dataframe Columns

In [20]:

```
data = {"UK": uk,
        "Russia": russia,
        "India" : india,
        "USA": usa,
        "Pakistan" : pakistan,
        "Bangladesh" : bangladesh,
        "Brazil":brazil,
        "China": china,
        "Date" : DATE
       }
```

In [21]:

```
type(data)
```

Out[21]:

```
dict
```

In [22]:

```
corona = pd.concat(data,axis = 1)
```

In [23]:

```
corona.set_index("Date", inplace = True)
```

In [24]:

```
corona.head()
```

Out[24]:

	UK	Russia	India	USA	Pakistan	Bangladesh	Brazil	China
Date								
2020-2-15	9.0	2.0	3.0	15.0	0.0	0.0	0.0	571.0
2020-2-16	9.0	2.0	3.0	15.0	0.0	0.0	0.0	830.0
2020-2-17	9.0	2.0	3.0	15.0	0.0	0.0	0.0	1287.0
2020-2-18	9.0	2.0	3.0	15.0	0.0	0.0	0.0	1975.0
2020-2-19	9.0	2.0	3.0	15.0	0.0	0.0	0.0	2744.0

In [25]:

```
corona.shape
```

Out[25]:

```
(435, 8)
```

In [26]:

```
type(corona)
```

Out[26]:

```
pandas.core.frame.DataFrame
```

Chceking for null values if any

In [27]:

```
corona.isnull().sum()
```

Out[27]:

```
UK            0
Russia        0
India         0
USA           0
Pakistan      0
Bangladesh    0
Brazil        0
China         0
dtype: int64
```

Converting date to Date time format

In [28]:

```
corona.index = pd.to_datetime(corona.index)
```

Finally ! We got the required format and countries!

We also need to pick up te date column

In [29]:

```
corona
```

Out[29]:

	UK	Russia	India	USA	Pakistan	Bangladesh	Brazil	China
Date								
2020-02-15	9.0	2.0	3.0	15.0	0.0	0.0	0.0	571.0
2020-02-16	9.0	2.0	3.0	15.0	0.0	0.0	0.0	830.0
2020-02-17	9.0	2.0	3.0	15.0	0.0	0.0	0.0	1287.0
2020-02-18	9.0	2.0	3.0	15.0	0.0	0.0	0.0	1975.0
2020-02-19	9.0	2.0	3.0	15.0	0.0	0.0	0.0	2744.0
...	...	...	...	...	...	...	...	...
2021-04-20	4393307.0	4718854.0	15609004.0	32536920.0	766882.0	727780.0	14050885.0	90159.0
2021-04-21	4395702.0	4727125.0	15924806.0	32602224.0	772381.0	732060.0	14122795.0	90167.0
2021-04-22	4398431.0	4736121.0	16257309.0	32669279.0	778238.0	736074.0	14172139.0	90182.0
2021-04-23	4401109.0	4744961.0	16602456.0	32736373.0	784108.0	739703.0	14238110.0	90190.0
2021-04-24	4403170.0	4753789.0	16951769.0	32789653.0	790016.0	742400.0	14308215.0	90201.0

435 rows × 8 columns

In [30]:

```
corona.to_csv("corona_dataset",header=False,index=False)
```

Bar Chart Race

In [31]:

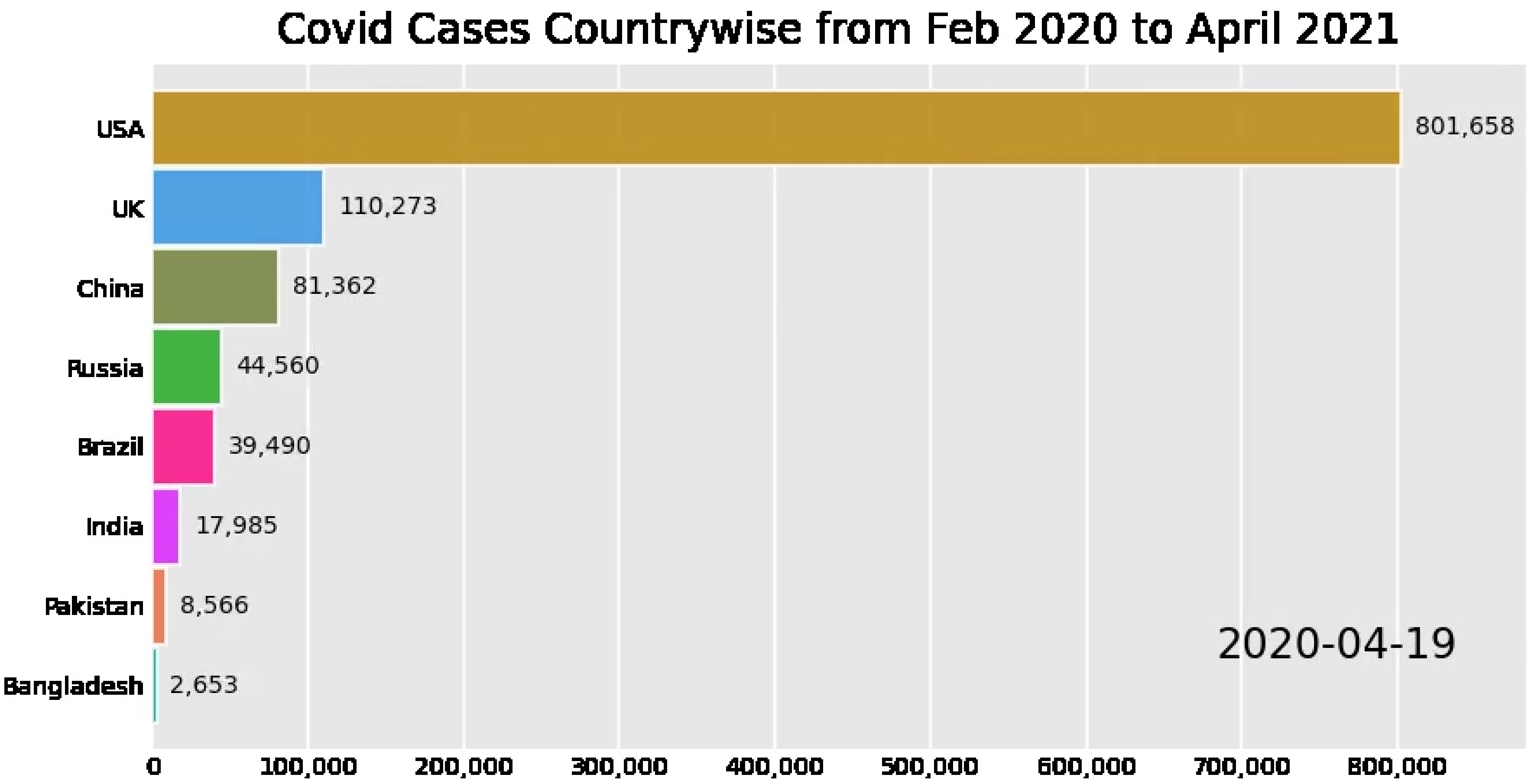
```
import bar_chart_race as bcr

bcr.bar_chart_race(df=corona,filename=None,title= "Covid Cases Countrywise from Feb 2020 to April 2021")
```

```
C:\Users\Avijeet\anaconda3\lib\site-packages\bar_chart_race\_make_chart.py:286: UserWarning: FixedFormatter should only be used to
gether with FixedLocator
  ax.set_yticklabels(self.df_values.columns)
```

```
C:\Users\Avijeet\anaconda3\lib\site-packages\bar_chart_race\_make_chart.py:287: UserWarning: FixedFormatter should only be used to
gether with FixedLocator
  ax.set_xticklabels([max_val] * len(ax.get_xticks()))
```

Out[31]:



In [ ]: